|  |  |  |  |
| --- | --- | --- | --- |
| **Lab Week 11 Report** | | | |
| **Class** | Database | **Group ID** | 11 |
| **Student ID** | 12151302 | **Name** | LEE HAESEONG |
| **Student ID** | 12151310 | **Name** | JANG HYOJUN |
| **Student ID** | 12171479 | **Name** | KIM HYUNJIN |
| **Student ID** | 12181465 | **Name** | KIM JAEWUK |
| **Lab Topic** | SQL Exercises with group members3 | | |

**Task 1: Display the list of all course sections offered in Spring 2010, along with the names of the instructors teaching the section.**

|  |
| --- |
| **Ⅰ. Code**  Select course\_id sec\_id ID,(case when name is NULL then ‘-’ else name end)as “instructor”  From (section natural left outer join teaches) Natural left outer join instructor  where semester =‘spring’ and year = 2010;  **Ⅱ. Explain**  Outer join은 join 조건에서 한쪽값이 없더라도 행을 반환한다. 즉  /Left table/ left outer join /right table/ 형식으로 사용되어야 하며 왼쪽에 있는 table은 조건에 부합하지 않더라도 모두 결합되어야 함을 뜻한다.  Outer join returns a row even if there is no one value in the join condition. Namely  /Left table/ left outer join /right table/ means that the table on the left should all be combined even if the conditions are not met.  spring,2010에 열렸던 section을 담당한 강사를 고르는 쿼리를 작성해야한다.  Make a query to select the instructor responsible for the section that was held in Spring,2010.  Section에 맞는 강사를 찾기 위해 teaches 묶어 조건을 탐색하여 하나의 table(left table)을 만들고 instructor 에서도 spring,2010에 열린 강의를 가르친 강사를 찾아야하므로 right table로 둔다.  To find instructors suitable for section, tie the tables together to create a table (actually left table) and keep them on the right table because the instructor also needs to find instructors who taught lectures in spring,2010.  즉 3개의 table에 대해서 같은 spring,2010 조건을 탐색하기위해 두번의 left outer join을 사용하였으며 on이 아닌 where을 사용해 결과에 spring,2010인 조건을 충족하는 것만 출력하도록 하였다.  In other words, to explore the same spring,2010 conditions for the three tables, two left outer joins were used and where instead of on was used to print only those that met the spring,2010 conditions in the results.  **Ⅲ. Result** |

**Task 2:** **Display the list of all departments, with the total number of instructors in each department, without using scalar subqueries.**

|  |
| --- |
| **Ⅰ. Code**  Select dept\_name,(count(ID)) as “Number of instructors”  From department natural left outer join instructors  Group by dept\_name;  **Ⅱ. Explain**  ‘Left outer join’ 의 ‘left table’로 ‘department’ 사용하여 모든 ‘department’가 출력되도록 해야 하며 on 없어도 ‘instructor’의 ‘dept\_name’이 ‘department’의 ‘dept\_name’를 참조하므로 결합할 수 있다. ‘dept\_name’은 여러 강사를 포함 할 수 있으므로 ‘count(ID)’를 이용해 각 ‘dept\_name’별로 강사의 ‘ID’ 를 세어 출력하는 쿼리를 작성했다.  The left table of the left outer join shall be used to print all department and can be combined without on, as the ‘dept\_name’ of the ‘instructor’ refers to the ‘dept\_name’ of the department.  Since dept\_name can include multiple instructors, a query was created using count(ID) to count and print the instructor ID for each dept\_name.    **Ⅲ. Result** |